# Acquisition of Conceptual Design Knowledge in Structural Engineering

M. Freischlad, T. Pullmann, M. Schnellenbach-Held, University Duisburg Essen, 45117 Essen, Germany
({mark.freischlad, torben.pullmann, m.schnellenbach-held}@uni-essen.de)

A. Albert, Krebs und Kiefer GmbH, 64295 Darmstadt, Germany
(alb@da.kuk.de)

## Summary

Expert systems integrating fuzzy reasoning techniques represent a powerful tool to support practicing engineers during the early stages of structural design. In this context fuzzy models have proved themselves to be very suitable for the representation of complex design knowledge. However their definition is a laborious task. This paper introduces an approach for the design and the optimization of fuzzy systems based upon Genetic Programming. To keep the emerging fuzzy systems transparent a new framework for the definition of linguistic variables is also introduced.

## 1 Introduction

Within the last 5 years EnvIOS Design, an expert system integrating fuzzy reasoning techniques has been created with the goal to support practicing engineers during the early stages of structural design. During the application of EnvIOS Design fuzzy models have proved themselves to be very useful for the modeling of complex structural design knowledge and especially for the formalization of conceptual design knowledge (Schnellenbach-Held et. al. 2002). However with respect to the acceptance of EnvIOS Design in practice the integrated fuzzy models have to be transparent and comprehensible. This fact dramatically complicates their automatic definition. Suitable methods for such an automatic definition are not yet available. In the research work described in this paper a new approach for the design and the optimization of fuzzy systems is introduced. This approach is based upon Genetic Programming (GP). To meet the requirement of transparency of the emerging fuzzy models special attention must be paid to the definition of the linguistic variables (i.e. to the definition of the fuzzy sets and the linguistic labels associated with these sets).

In the second chapter of this paper a new framework for the definition of linguistic variables called the Individual Linguistic Label (ILL) model is introduced. This framework allows for high flexibility concerning the number of fuzzy sets in a linguistic variable and for certain variations concerning their shapes. Thus a very high accuracy of a certain fuzzy model can be reached. At the same time a particular linguistic variable can be used within several fuzzy models without changing the meaning of the single linguistic labels i. e. without changing the fuzzy sets associated with them. A good interpretability of the single linguistic labels is guaranteed even across different fuzzy models. The latter is particularly important for the formalization of conceptual design knowledge where one linguistic variable is often used within several fuzzy models.

The third chapter introduces a new approach for the design and the optimization of fuzzy systems based on Genetic Programming making use of the Individual Linguistic Label (ILL) model.

## 2 Transparent fuzzy models for knowledge representation

For Representation of knowledge within an expert system, rule-based fuzzy systems (RBFS) have been proven as suitable solution. In this context Takagi / Sugeno / Kang (TSK) models (Takagi & Sugeno 1985) and also Mamdani / Assilian (MA) models (Mamdani 1975) have been used (Albert 2002). Without loss of generality, the MA-model will be considered in further descriptions.

### 2.1 Descriptive vs. approximative modeling

Rule-based fuzzy models can be spread into two different modeling approaches. A model contains rules in the form:

$$R_i : If\ x_1\ is\ A_{i1}\ and\ ...x_n\ is\ A_{in}\ then\ y\ is\ B_i \quad (1)$$

In the descriptive model $A_{ij}$ ($B$) is one of the predefined fuzzy sets of the input variable $x_j$ (output variable $y$). In general linguistic labels are assigned to these fuzzy sets (figure 1a). In the approximative model $A_{ij}$ ($B$) are individual fuzzy sets of the input (output) variables. Figure 1b shows it is often impossible to assign linguistic labels to these fuzzy sets.

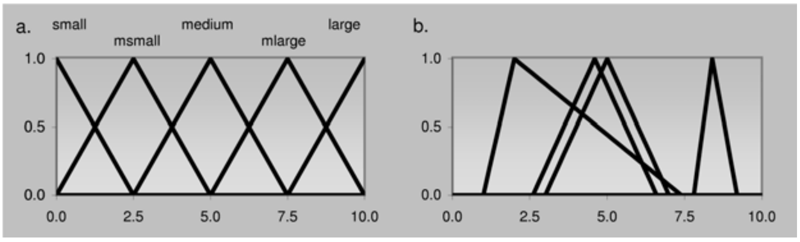


Figure 1: Fuzzy sets of (a.) descriptive and (b.) approximative models

Main advantage of the approximative model is the higher level of accuracy. Due to a limited transparency this model is not feasible for knowledge representation. In contrast to this the descriptive model has a very good interpretability. Otherwise the definition of fuzzy sets turns out to be very complex. To keep the interpretability the adjustments of these sets are limited.

Within an expert system fuzzy variables are being used in different fuzzy systems. For instance the span length of a concrete slab influences the decision of a suitable slab system as well as the determination of suitable element dimensions (Albert 2002). This additionally aggravates the tuning of the fuzzy sets. To reach an appropriate accuracy it is mostly necessary to use a high number of fuzzy sets. This results in a high number of used rules, which finally leads to a loss of interpretability. Thus in terms of transparency of the knowledge representation the usual descriptive model is only limited feasible.

### 2.2 Individual linguistic label model

For representation of complex engineering knowledge within EnvIOS Design the "individual linguistic label" (ILL) model has been developed. This model combines the advances of the descriptive and the approximate model. The $A_{ij}$ ($B$) are Gaussian fuzzy sets of the generic form:

<operator> <hedge$_1$> <prototype$_1$> (<hedge$_2$> <prototype$_2$>) (2)

The Gaussian deviation parameter *sigma* and the possible prototypes are predefined for each variable. The membership function can be modified by the dilation hedges "about" or "roughly" (figure 2a). Possible operators are "between" (figure 2b), "at most" and "at least" (figure 2c).

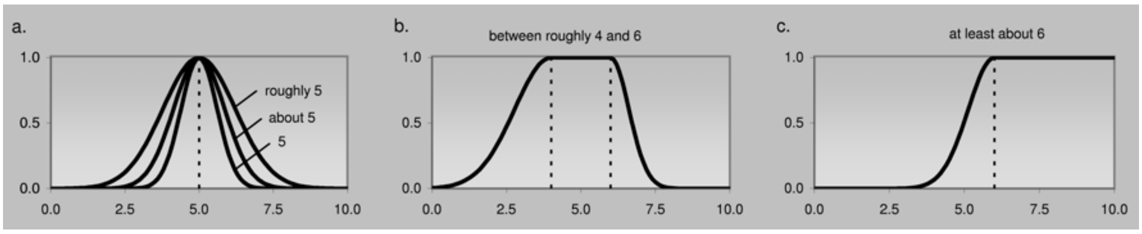


Figure 2: Individual linguistic labels

Furthermore another benefit of the ILL model for knowledge representation within an expert system is the very good interpretability of single rules. A major characteristic of expert systems is the description component. This component enables the user to trace the proceedings of the expert system and shows important explanations. A single rule of a common descriptive model can only be interpreted in the context of the whole rule base. Consequently the description component has to show the whole rule base. In the case of complex fuzzy systems this violates the demand for transparency. In case of the ILL model only rules which had a significant impact on the results have to be shown.

# 3 Genetic programming based design of fuzzy systems

The design of fuzzy systems is a very complex task. Suitable methods for an automatic definition are not available yet. In this chapter a new approach for the design and the optimization of fuzzy systems is introduced. This approach is based upon Genetic Programming and takes into account the requirement of transparency of the resulting fuzzy models.

## 3.1 Evolutionary Algorithms

Evolutionary Algorithms (EA) are universal optimization algorithms which emulate the principles of the natural evolution: selection, reproduction and mutation (Goldberg 1989). The attributes of an object that should be optimized are represented by the genome. First a population of $N_{pop}$ individuals with random genomes is initialized. In the selection process the fitness of each individual is evaluated using a predefined fitness function. The offspring population is generated by crossing and mutating individuals of the parent generation, whereas individuals with a higher fitness are used with a higher probability. This process is repeated for $N_{gen}$ generations.

The most common EA are genetic algorithms (GA), evolutionary strategies and genetic programming (GP).

## 3.2 Genetic Programming

There is a main difference between GP and an ordinary GA. Genetic Algorithms represent an individual through binary or numerical strings with mostly fixed length. Genetic Programming represents the individuals in form of complex structures, mostly trees. The Individuals do not have to follow a fixed size or a certain pattern. This leads to a greater diversity and variability of GP individuals (Koza 1992).

A major advance of GP algorithms is their excellent performance at crossover operations. GA algorithms use random crossover points within their string representation, without respect to groups or substructures. It is hardly predictable if good substructures stay together or are randomly destroyed. GP algorithms use certain node elements within the tree for crossover, so the probability of keeping good substructures together is much higher.

## 3.3 DA-GPFS

The *D*omain Knowledge *A*ugmented *G*enetic *P*rogramming based *F*uzzy *S*ystem (DA-GPFS) is a genetic fuzzy system (Bodenhofer & Herrera 1997) for the data-driven generation of fuzzy rule based systems by means of genetic programming. Fuzzy systems are represented by a tree structure according to the Backus-Naur-Form (BNF) (Geyer-Schulz 1995). The fitness function takes into account the accuracy of the fuzzy system by means of the mean squared approximation error (MSE) as well as the interpretability by panelizing large size of the rule base and the length of the rules. The algorithm incorporates domain specific knowledge that is used by human knowledge engineers in the manual fuzzy system design process.

### 3.3.1 Directed mating

The combination of characteristics of two individuals (crossover) is most interesting if both individuals complement each other. In terms of fuzzy systems this means a potential crossover partner (parent B) has a lower approximation error in an area of the input space, where parent A has a higher error. The potential mean error of parent *A* and candidate mate *m* is evaluated by

$$PME_{Am} = \sum_{j=1}^{Nds} \mathrm{MIN}(E_{Aj}, E_{mj}) / Nds \tag{3}$$

where *Nds* is the number of input vectors and $E_{Aj}$ ($E_{mj}$) is the absolute approximation error of parent *A* (mate *m*) and the *j*th input vector.

In a standard evolutionary algorithm crossover mates are selected randomly. Beside this standard method a method for directed mating is available in DA-GPFS:

- Random selection of parent *A*
- Random selection of $N_{CM}$ candidate mates
- Evaluation of $PME_{Am}$ for each candidate mate
- Selection of the candidate mate with the minimal *PME* as parent *B*

### 3.3.2 Goal-oriented evolutionary operators

The combination of two fuzzy systems should combine “good” rules of both parents. In contrast to this the manipulation (mutation) should change the characteristics of “bad” rules. Rules are normally considered as “good”, if training data within their diffuse within the input space are approximated unusually well.

The fitness of rule *i* is evaluated by

$$RF_i = \sum_{j=1}^{Nds} (SE_j \cdot \mu'_{rule,ij}) / \sum_{j=1}^{Nds} \mu'_{rule,ij} \tag{4}$$

where

$$\mu'_{rule,ij} = \mu_{rule,ij} / \sum_{i=1}^{Nr} \mu_{rule,ij} \tag{5}$$

and $\mu_{rule,ij}$ is the degree of confidence of the *i*th rule and the *j*th input vector, *Nr* is the number of rules, $SE_j$ is the squared approximation error of the *j*th input vector.

In a genetic programming algorithm with a tree structure representation each node with its attached sub tree represents a substructure. The crossover is carried out by replacing a randomly selected substructure of parent *A* by a randomly selected substructure of parent *B*. An individual is mutated by replacing a randomly selected sub tree by a randomly generated tree according to the representation language, i.e. the Backus-Naur-Form. In the DA-GPFS algorithm the substructures for crossover and mutation, respectively, are determined by a roulette wheel selection in consideration of the substructure fitness. The rule fitness *RF* is directly assigned to the rule nodes and nodes with a lower hierarchy level, i.e. Conditions and Conclusions. Nodes with a higher level, the rule base nodes, get the mean rule fitness of the attached rules as substructure fitness (figure 3).

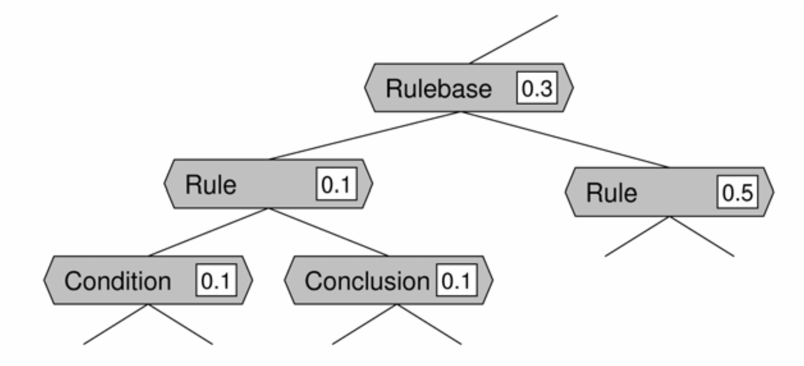


Figure 3: Attribution of substructure fitness

# 4 Applications

The models presented in this paper were applied to practical problems in the frame of knowledge based systems in structural engineering using the expert system environment EnvIOS Design. This environment is based on the OFWM, a knowledge representation model for the formalization of conceptual and detailed design knowledge.

## 4.1 The knowledge representation model

The Object Oriented Fuzzy Knowledge Representation Model (Objekt-orientiertes Fuzzy-Wissensrepräsentations-Modell, OFWM) (Albert 2002) consists of two models the Product Model (Produktmodell, PM) and the Model of Knowledge Base Elements (Modell der Elemente der Wissensbasis, MEW). The Product Model serves to organize the elements of a knowledge base. The Model of Knowledge Base Elements serves for the formalization of the design knowledge.

The elements of a design calculation are the variables that either can be found in design standards or belong to the knowledge of an experienced structural engineer. In the context of the OFWM these elements are called “calculation elements”. They are separated into different classes according to the method that they use in order to determine their own values. Thus calculation elements can be of types like “formula”, “table”, etc.

During the design of concrete structures several complex decisions have to be made. Besides the choice of suitable structural components the necessity of proofs for the components should be checked. There are not all proofs required for all components. In order to maintain the transparency only essential proofs should be brought forward. The MEW includes the fuzzy logic based knowledge elements MA- and TSK-model for the representation of preliminary design knowledge. Complex knowledge about the necessity of proofs can be represented by the TSK-premise knowledge element.

## 4.2 Software applications

The applications “**Env**ironment for **I**ntelligent **O**bject oriented **S**tructural **Design**” (EnvIOS Design) and “**K**nowledge **B**ase **D**efinition **T**ool” (KBDT) were developed for the implementation and use of OFWM based conceptual and structural design applications. EnvIOS Design consists of two modules. EnvIOS Design I, merged in the AutoCAD environment, is the user interface for defining, editing and displaying structure models, which are stored according to the product model scheme. EnvIOS Design II contains the inference machine, where OFWM knowledge bases are applied to structure models. The KBDT represents the knowledge acquisition component of EnvIOS Design. Using the KBDT OFWM-knowledge bases can be defined and edited. Based on the DA-GPFS algorithm the new KBDT module GPFuzzyStudio was implemented.

## 4.3 First results

In order to examine the suitability of the developed models DA-GPFS was applied to the data-driven generation of fuzzy systems for the preliminary design of slab systems. In first experiments a fuzzy system for the determination of the beam height of a girder slab was generated. The generated fuzzy system, consisting of eight transparent rules according to the ILL model, delivered a mean approximation error of less than 2.5 % on the training data set and less than 4 % on the test data set. These experiments have shown that the convergence speed was seriously increased by the domain knowledge augmented genetic operators.

# 5 Conclusions

A new approach for transparent representation of engineering knowledge was presented. The Individual Linguistic Label Model combines the advantages of the approximative and the descriptive model. Because of the good interpretability of single rules the model is predestined for the application within a fuzzy expert system. In addition a genetic programming based approach for the data-driven generation of fuzzy systems was presented. The DA-GPFS includes sophisticated crossover and mutation methods that increase the convergence speed of the optimization process.

# 6 References


Albert, A. (2002): Wissensbasiertes Entwerfen und Bemessen von Tragwerken unter Einsatz von Fuzzy-Methoden, Dissertation, TU Darmstadt, published as: Fortschritt Bericht VDI, Reihe 4, Nr. 179, VDI-Verlag, Düsseldorf.

Bodenhofer, U., Herrera, F. (1997): Ten Lectures on Genetic Fuzzy Systems, Preprints of the International Summer School: Advanced Control – Fuzzy, Neural, Genetic, R. Mesiar (Ed.), Slovak Technical University, Bratislava.

Geyer-Schulz, A. (1995): Fuzzy Rule-Based Expert-Systems and Genetic Machine Learning. Physica-Verlag.

Goldberg, D. E. (1989): Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley Publishing Company, Reading, Massachusetts.

Koza, J. R. (1992): Genetic Programming: On the Programming of Computers by Means of Natural Selection. Cambridge, MA, MIT Press.

Mamdani, E. H., Assilian, S. (1975): An Experiment in Linguistic Synthesis with a Fuzzy Logic Controller, International Journal of Man-Machine Studies, 7.

Schnellenbach-Held, M., Freischlad, M., Albert, A. (2002): Fuzzy Rule Based Models for Structural Design Decisions, Proceedings of IABSE-Symposium – Towards a Better Built Environment / Innovation, Sustainability, Information Technology, Melbourne.

Takagi, T., Sugeno, M. (1985): Fuzzy identification of systems and its application to modeling and control. IEEE Transactions on Systems, Man, Cybernetics. 15(1).